

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A method performed by a computer for computing modified discrete cosine transform of an input signal $y(k)$, the method comprising the steps of:

receiving the input signal $y(k)$;

computing re-arranging data of the input signal $y(k)$, as

$$x(k) = \begin{cases} [-y(26-k) - y(27+k)] \cdot b_k & \text{for } 0 \leq k \leq 8 \\ [y(k-9) - y(26-k)] \cdot b_k & \text{for } 9 \leq k \leq 17 \end{cases};$$

computing $Y'(n) = \sum_{k=0}^{17} x(k) \cos[\frac{\pi}{36}(2k+1)n]$ for $0 \leq n \leq 17$;

defining $Y(0) = Y'(0)/2$; and

computing generating an output signal $Y(n)$ according to

$$Y(n) = Y'(n) - Y(n-1) \text{ for } 1 \leq n \leq 17,$$

where ~~y is an input data~~, $x(k)$ is re-arranged data for ~~y~~ the input signal $y(k)$, Y' is discrete cosine transform of x , output signal Y is the modified discrete cosine transform of the input signal y , and b_k is a constant.

2. (Currently Amended) An MP-III encoder/decoder comprising:
means for receiving an input signal $y(k)$;

Appln No. 09/780,699

Amdt date March 25, 2005

Reply to Office action of January 7, 2005

means for ~~computing~~ re-arranging data of the input signal $y(k)$, as

$$x(k) = \begin{cases} [-y(26-k) - y(27+k)] \cdot b_k & \text{for } 0 \leq k \leq 8 \\ [y(k-9) - y(26-k)] \cdot b_k & \text{for } 9 \leq k \leq 17 \end{cases};$$

means for computing $Y'(n) = \sum_{k=0}^{17} x(k) \cos[\frac{\pi}{36}(2k+1)n]$ for $0 \leq n \leq 17$;

means for defining $Y(0) = Y'(0)/2$; and

means for ~~computing~~ generating an encoded output signal $Y(n)$ according to

$$Y(n) = Y'(n) - Y(n-1) \text{ for } 1 \leq n \leq 17 ,$$

where ~~y is an input data~~, $x(k)$ is re-arranged data for ~~y~~ the input signal $y(k)$, Y' is discrete cosine transform of x , output signal Y is the modified discrete cosine transform of the input signal y , and b_k is a constant.

3. (Currently Amended) The encoder/decoder of claim 2, further comprising:

means for receiving the encoded signal $Y(k)$;

means for computing $Y''(k) = Y(k) \cdot b_k$ for $0 \leq k \leq 17$;

means for computing $y'''(n) = \sum_{k=0}^{17} Y''(k) \cos[\frac{\pi}{2 \cdot 18}(2k+1)n]$ for $0 \leq n \leq 17$;

means for computing $y'(n) = \begin{cases} y'''(n+9) & \text{for } 0 \leq n \leq 8 \\ 0 & \text{for } n = 9 \\ -y'''(27-n) & \text{for } 10 \leq n \leq 26 \\ -y'''(n-27) & \text{for } 27 \leq n \leq 35 \end{cases};$

means for defining $y(0) = \sum_{k=0}^{18-1} Y(k) \cdot c_k$; and

Appln No. 09/780,699

Amdt date March 25, 2005

Reply to Office action of January 7, 2005

means for generating a decoded output signal $y(n)$ according to

$$y(n) = y'(n) - y(n-1) \text{ for } 1 \leq n \leq 35,$$

where Y'' is the modified discrete cosine transform of the encoded signal $Y(k)$ ~~y~~ multiplied by b_k , y''' is the discrete cosine transform of Y'' , and y' is re-arranged data for y''' .

4. (Currently Amended) An electronic circuit for fast computation of computing modified discrete cosine transform comprising:

a receiving circuit for receiving an input signal $y(k)$;

a first circuit for ~~computing~~ re-arranging data of the input signal $y(k)$, as

$$x(k) = \begin{cases} [-y(26-k) - y(27+k)] \cdot b_k & \text{for } 0 \leq k \leq 8 \\ [y(k-9) - y(26-k)] \cdot b_k & \text{for } 9 \leq k \leq 17 \end{cases};$$

a second circuit for computing $Y'(n) = \sum_{k=0}^{17} x(k) \cos[\frac{\pi}{36}(2k+1)n]$ for $0 \leq n \leq 17$;

a third circuit for defining $Y(0) = Y'(0)/2$; and

a fourth circuit for ~~computing~~ generating an output signal $Y(n)$ according to

$$Y(n) = Y'(n) - Y(n-1) \text{ for } 1 \leq n \leq 17,$$

where ~~y is an input data~~, $x(k)$ is re-arranged data for ~~y~~ the input signal $y(k)$, Y' is discrete cosine transform of x , output signal Y is the modified discrete cosine transform of the input signal y , and b_k is a constant.

Appln No. 09/780,699
Amdt date March 25, 2005
Reply to Office action of January 7, 2005

5. (Currently Amended) A method performed by a computer for computing modified inverse discrete cosine transform of an input signal Y(k), the method comprising the steps of:

receiving the input signal Y(k);

computing $Y''(k) = Y(k) \cdot b_k$ for $0 \leq k \leq 17$;

computing $y'''(n) = \sum_{k=0}^{17} Y''(k) \cos[\frac{\pi}{2 \cdot 18} (2k+1)n]$ for $0 \leq n \leq 17$;

computing

$$y'(n) = \begin{cases} y'''(n+9) & \text{for } 0 \leq n \leq 8 \\ 0 & \text{for } n = 9 \\ -y'''(27-n) & \text{for } 10 \leq n \leq 26 \\ -y'''(n-27) & \text{for } 27 \leq n \leq 35 \end{cases} ;$$

defining $y(0) = \sum_{k=0}^{18-1} Y(k) \cdot c_k$; and

computing generating an output signal y(n) according to

$$y(n) = y'(n) - y(n-1) \text{ for } 1 \leq n \leq 35 ,$$

where Y'' is the modified discrete cosine transform of ~~y~~ the input signal Y(k) multiplied by b_k , y''' is the discrete cosine transform of Y'' , and y' is re-arranged data for y''' .

6. (Currently Amended) An electronic circuit for fast computation of ~~computing~~ modified inverse discrete cosine transform of an input signal Y(k) comprising:

a receiving circuit for receiving the input signal Y(k);

a first circuit for computing $Y''(k) = Y(k) \cdot b_k$ for $0 \leq k \leq 17$

Appln No. 09/780,699

Amdt date March 25, 2005

Reply to Office action of January 7, 2005

a second circuit for computing $y'''(n) = \sum_{k=0}^{17} Y''(k) \cos\left[\frac{\pi}{2 \cdot 18} (2k+1)n\right]$ for $0 \leq n \leq 17$

a third circuit for computing

$$y'(n) = \begin{cases} y'''(n+9) & \text{for } 0 \leq n \leq 8 \\ 0 & \text{for } n = 9 \\ -y'''(27-n) & \text{for } 10 \leq n \leq 26 \\ -y'''(n-27) & \text{for } 27 \leq n \leq 35 \end{cases}$$

a fourth circuit for defining $y(0) = \sum_{k=0}^{18-1} Y(k) \cdot c_k$; and

a fifth circuit for ~~computing~~ generating an output signal $y(n)$ according to

$$y(n) = y'(n) - y(n-1) \text{ for } 1 \leq n \leq 35,$$

where Y'' is the modified discrete cosine transform of ~~y~~ the input signal $Y(k)$ multiplied by

b_k , y''' is the discrete cosine transform of Y'' , and y' is re-arranged data for y''' .